Cognitive Binding: A Computational-Modeling Analysis of a Distinction between Implicit and Explicit Memory

Janet Metcalfe
Dartmouth College

Garrison W. Cottrell
University of California, San Diego

W. E. Mencl
Dartmouth College

Abstract

■ Four models were compared on repeated explicit memory (fragment cued recall) or implicit memory (fragment completion) tasks (Hayman & Tulving, 1989a). In the experiments, when given explicit instructions to complete fragments with words from a just-studied list—the explicit condition—people showed a dependence relation between the first and the second fragment targeted at the same word. However, when subjects were just told to complete the (primed) fragments—the implicit condition—stochastic independence between the two fragments resulted. Three distributed models—CHARM, a competitive-learning model, and a back-propagation model produced dependence, as in the explicit memory test. In contrast, a separate-trace model, MINERVA, showed independence, as in the implicit task. It was concluded that explicit memory is based on a highly interactive network that glues or binds together the features within the items, as do the first three models. The binding accounts for the dependence relation. Implicit memory appears to be based, instead, on separate noninteracting traces. ■

INTRODUCTION

One function of computational models is to allow us to analyze patterns of human data that are consistent but seem theoretically opaque. The patterns explored here are the findings that explicit-memory tasks show a dependence relation to one another, whereas, implicit-memory tasks show stochastic independence. Our question is: What do these dependence/independence results mean about the human memory systems?

The particular task that we shall investigate is repeated fragment completion. This is an especially interesting task from a modeling perspective because one of the strong points of interactive network models is that they are capable of specifying, in detail, a mechanism whereby pattern or fragment completion can be enacted. So the question that we ask here is not whether a model can do the task at all. It is, rather, whether it produces the pattern of contingencies on repeated testing that humans reveal in the implicit or in the explicit task. By analyzing the models that show the implicit pattern and those that produce the explicit pattern we can tease apart the fundamental characteristics of those two systems that give rise to the differing patterns of results. We shall briefly review the argument that there are separable memory systems. The modeling results presented here, though, go beyond the idea that the systems may be separable and begin to explore *how* they differ and *what the underlying principles in each of the systems might be.*

BACKGROUND: DIFFERENT MEMORY SYSTEMS?

A number of theorists have proposed that there may be (at least) two functionally distinct memory systems sometimes called episodic—or True Memory, and semantic—or Quasi Memory (Tulving, 1986), that may be differentially tapped by explicit and implicit tasks. The episodic system is assumed to store events that (1) have place and time knowledge, (2) have some personal meaning to the individual as having occurred to him or her, specifically, rather than just being world knowledge, (3) are intimately hooked up with a functional hippocampal memory system, and, thus, susceptible to certain kinds of

amnesia, and (4) require conscious recollective processes for their encoding and retrieval. The Quasi Memory system may show some plasticity, and, in particular, priming effects may manifest themselves, but there need be no conscious recollection associated with this system, or with implicit remembering.

Evidence for two kinds of processing or systems has been reviewed by Richardson-Klavehn and Bjork (1988), and others. First, a number of variables (such as "level of processing," or whether or not a subject generates, as compared to reads, a word) influence explicit tasks such as recall and recognition, but cause no differential effect on implicit tasks such as lexical decision, stem, or fragment completion (e.g., Jacoby & Dallas, 1981; Graf, Mandler, & Haden, 1982). Second, explicit tasks, like free recall, cued recall, and recognition, are selectively impaired in amnesics as compared to normals or matched control patients; performance on the implicit tasks, such as primed fragment completion, stem completion and lexical decision is spared (Shimamura, 1986; Schacter, 1987). The third line of evidence is contentious. It was initially thought that the tasks within a given system would show dependence with one another, whereas independence across systems would be found. Many experiments show that two classic explicit tasks—recognition and recall—are dependent, such that the probability of recall of a particular target is higher given the subject recognized it than had the subject failed to recognize it.

The problem with the initial interpretation of the meaning of dependence and independence comes from studies that flagrantly violate intuitions based on this kind of reasoning. Although implicit memory tasks have been shown to be independent of explicit memory tasks—fragment completion is independent of recognition (Tulving, Schacter, & Stark 1982)—different implicit tasks are also independent of one another. Witherspoon and Moscovitch (1989), for example, found that word identification was independent of fragment completion even though both are presumably enacted by the same system. We are left in a quandary about what dependence and independence means. This situation was taken to its logical extreme by Hayman and Tulving (1989a), who conducted repeated fragment completion tests with exactly the same task and targets, using only different parts of the same words as cues. In the explicit memory version of the tasks they found dependence, but in the implicit version independence was found.

We share Witherspoon and Moscovitch's (1989) reservations about having to postulate separate systems for each and every test that is shown to be independent of some other. This seems implausible, especially when the tests are trivial variants of one another (e.g., both are fragment completion tests directed at one and the same target).

Roediger, Weldon, and Challis (1989) have suggested that the transfer appropriate processing view may be called for. This approach relies on a careful assessment of the compatibility relations between encoding and test, and analyses of the processes involved. They note, however, that the systems view and the transfer appropriate processing view may not, in fact, be at odds with one another, saying: "We must admit that there is no inherent reason that an approach specifying both memory systems and something like processing modes or procedures cannot be partially correct. Neural structures require processing for their operation, and procedures must be carried out by the brain. A theory specifying both structural bases and processing assumptions is needed (Anderson, 1978), but those presently on the scene emphasize either structure to the relative neglect of processing assumptions (the systems approach) or processing assumption to the relative neglect of structure (our own approach)" (Roediger et al., 1989, p. 36).

We take this third approach—specifying both the systems and structure and the processes involved. Working models could not function otherwise. Before describing the models and their application to the implicit/explicit distinction, however, it is necessary to provide a few more details about the paradigm we examine and the experimental results.

THE EXPERIMENTAL PARADIGM

Hayman and Tulving (1989a) investigated the dependency relations in fragment completion and cued fragment recall in a series of four experiments. Typically, subjects were presented a sequence of words in a study list containing some words, such as AARDVARK, which would later be the primed targets. Later, subjects were given one of two tests—an *implicit* test, in which they were asked to complete fragments, such as _AR_VA_ _, or a cued-recall (i.e., *explicit*) test in which they were asked to recall the word from the list that contained the following letters: _AR_VA_ _. In both cases, there was a second test in which the subject was either given the same fragment a second time or a different complementary fragment. So, for example, the second complementary fragment for aardvark would be A_ _D_ _RK.

In the control cases, in which the exact same fragments were used on the second test, no matter what the instructions to the subjects were, strong dependence between the first and second test was exhibited. This makes sense. If the tests are exactly the same then the only difference should be due to the noise of testing itself, including possible interfering effects of intervening items and possible priming effects due to the first test. In the models, we need not implement these confounding factors. In the exact-repetition case, obviously, all models would show complete dependence. When fragments are not identical, those that have more letters in common should show more dependence than those without letters in common. The limiting case for showing minimal depen-

dence is no letters in common. The limiting case for maximal dependence is exact repetition of all letters.

A number of researchers have pointed up potential problems with the repeated testing methodology (see Shimamura, 1985), as well as some solutions (Flexser, 1981; Hayman & Tulving, 1989b). To circumvent these problems, Hayman and Tulving (1989a) used a method of testing called the "reduction method" (Watkins & Todres, 1978) that compares a control list not involving repeated testing with the conditionalized list. It is designed to offset the effects of repeated testing. In the formal modeling such a remedy for the priming effects due to the test situation, will, of course, not be needed, since we do not alter the models when testing them.

Hayman and Tulving's (1989a) results were as follows: (1) In both the implicit- and the explicit-memory tasks, when the exact same fragment was repeated, dependence between the two tests was obtained. It would be cause for considerable concern had this not been the case, and the result is predicted by all models of memory. (2) In the different fragment condition, with the explicit-memory task, dependence was obtained. (3) In the implicit-memory conditions, in the different fragment condition, the results of the two tests were independent of one another. The challenge, then, is to say what kinds of systems give rise to the dependence seen in the explicit task and what kinds of systems can produce the independence seen in the implicit task.

MODELS

Three of the models we will report upon are "distributed." In these models the features representing the items are modified by one another by the storage mechanisms in the postulated networks. One of the models (MINERVA, Hintzman, 1987) was a separate multiple-trace model. In this model, both the items and the features within the items are simply stored independently.

In deciding on what representations to use for the fragments, we rediscovered the wheel several times. The gist of the results was the same for all four models so rather than reporting multiple simulations, these results are summarized here. First, we drew the features randomly with replacement to be in the first or the second fragment. Consider the case where 0.5 of the features are in fragment 1 and 0.5 are in fragment 2. If these are randomly selected, without respect for one another then we would expect that the two fragments will have $0.5 \times 0.5 = 0.25$ features in common. The common subset of the features induces a positive correlation. We found that with this coding scheme for the fragments, all of the models we tested showed positive dependence.

Second, we made the features in the two fragments mutually exclusive. If there were no overlap among the features in the two fragments to artifactually induce a correlation, then the intrinsic characteristics of the model should shine through. If the model was inherently independent under fragment completion, then that independence was expected to show up when the fragments were mutually exclusive. If the mechanisms or connections in the model itself induced some dependency then that too should be manifested. In our first attempts to make the features mutually exclusive, however, we accidentally induced a negative tradeoff between the two fragments. We went through the item, feature by feature, flipping a computer-generated fair coin for each feature. If a feature was assigned to one fragment, the other fragment was given a value of zero on that feature. The problem with doing this is that there is considerable variability in flipping the coin (and in the underlying information content of each feature). If one fragment accidentally got 60%, rather than 50% of the features, then the other fragment necessarily got only 40%. Thus, fragments that were "good" fragments, i.e., had an above average number of features, were necessarily linked to "bad" fragments, and this produces a negative correlation a priori. All of the models showed less dependence under these conditions than with independent feature sampling. Some of the models (MINERVA and backpropagation), under some parameter values, actually demonstrated a negative dependence relation under these conditions. Because the correlation or lack thereof was due to idiosyncrasies of the coding rather than to the intrinsic characteristics of the model, we decided that this scheme was not a fair test of the models. To demonstrate the intrinsic characteristics of the model, then, we settled on a mutually exclusive sampling scheme that minimizes the negative tradeoff described above. The scheme was extremely simple—the first half of the features comprised one fragment and the second half the other.

CHARM Model

The CHARM model was originally designed to address the question of how people associate, store, and retrieve events in True Memory system (Metcalfe, 1985, 1991, 1992). Items, represented as multidimensional vectors, are associated by the operation of convolution, and are stored by being added into a composite memory trace, itself a vector, along with other such convolved pairs of items. In the simulation described below the items are autoassociated. At time of retrieval, the cue vector is correlated with the composite memory trace. This produces a vector that is identified by being matched to every item in a lexicon. The best-matching item, as long as that item exceeds a lower criterion of goodness-of-match, is considered to be the item generated. An overview of the model is shown in Figure 1.

Method

Lexicons of 100 vectors, each consisting of 93, 63, or 33 features, which were numbers randomly sampled from

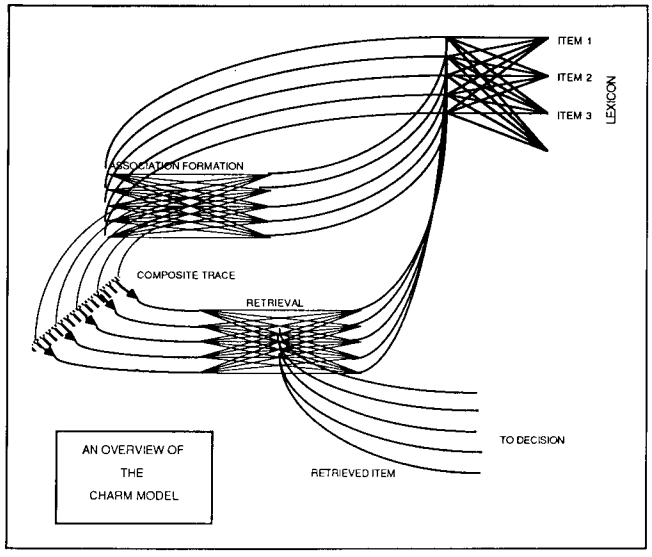

Figure 1. An overview of the CHARM model.

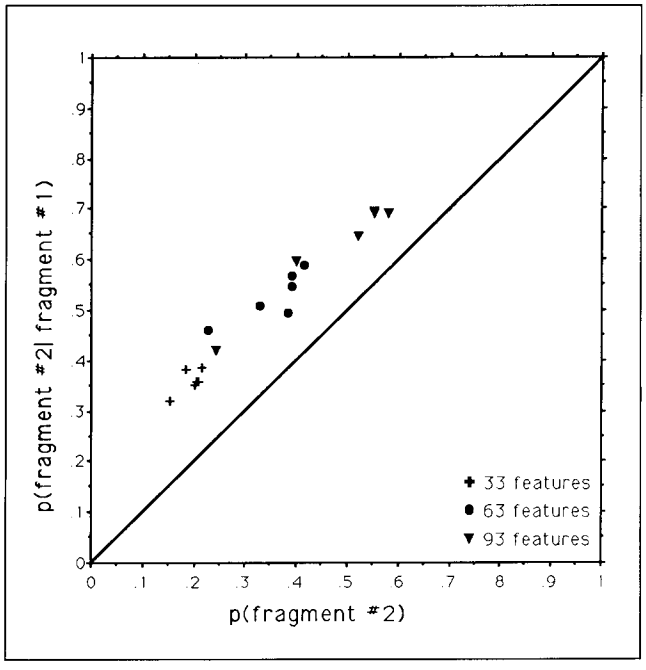

Figure 2. The repeated fragments dependence relation in CHARM.

a truncated normal distribution centered around a value of 0, were constructed. Because of this sampling procedure, the items in the lexicon were statistically independent of one another and could be said to be psychological entities like unrelated words. The first 12 items of the lexicon were autoassociated, by the operation of convolution, and added into the composite memory trace.

The first cue was constructed for each of the 12 items by selecting the first half of the features to comprise the cue, and the remainder to be coded as zeros. These fragments were correlated with the composite memory trace and the retrieved item was matched to every item in the lexicon by taking the dot product to each. The criterion for saying that the resonance between the retrieved item and the highest resonating lexical item was good enough for retrieval to be said to have occurred was varied from 0 to 0.2, 0.4, 0.6, 0.8, to 1. The item that had the highest dot product, as long as it was above the criterion for that particular run, was said to be the item that was given as the completion to the fragment. For the second test, the fragment cues were the last half of the features in the item, with the other elements being coded as zeros.

Whether the simulation produced the correct result to the first and/or second fragment was tabulated in a two by two contingency table, from which the simple and conditional probabilities were computed. Each treatment combination was run through 100 independent runs, so the data points are each based on 1200 observations.

Results

The results are plotted in Figure 2, which shows the conditional probability of correct completion of fragment two given correct completion of fragment one on the ordinate, plotted against the simple probability of correct completion of fragment 1, on the abscissa. The diagonal indicates independence; anything above the diagonal indicates positive dependence; points falling in the area below the diagonal would indicate negative dependence. As can be seen from the figure, there was a considerable amount of dependence produced by the model. Dependence was also produced in the experimental situation, but only under explicit conditions, i.e., where subjects were told to complete the fragment with a just-studied word from the list.

Discussion

The reason the CHARM model, under autoassociation, predicts dependence on the two tasks is that every feature is linked to every other feature, by the associative operation. When any feature is given as a cue or part of a cue at time of retrieval, it provokes signal terms from *all* the features in the initial representation, proportional to the absolute value of the cue feature. Since any feature produces a degraded representation of all of that item's features, a correlation among features is necessarily built into the autoassociative model.

The fact that the encoding is distributed or scattered in CHARM such that all features are linked to all is at the base of this correlation. (The operation of convolution, itself, spreads or smears the values of the features in this distributed manner.) As such, we should find that other models that share or partially share this kind of interactive representational coding such that some or all of the

features are linked, bound, or connected to all of the other features, should also share the dependence relation under conditions of repeated fragment completion, even in the case of mutually exclusive fragments. A particularly clear case is given by the competitive-learning model.

Competitive-Learning Model

The origins of the competitive learning procedure go back to work by von der Malsburg (1973) and Grossberg (1970). Rumelhart and Zipser (1986) extended these procedures, and provided corrections to problems in the original models. At heart, this type of model (see Fig. 3) consists of at least two layers of units connected by modifiable weight links. When a pattern of input units at the first layer is activated, units in the next layer "compete" for the privilege of responding to that particular pattern by inhibiting other units in the layer from becoming activated. Only the winning unit is allowed to modify its links, and does so in a manner that makes it increasingly more responsive to that particular input pattern.

Method

A competitive-learning model was implemented, which consisted of two layers of units, an input and a category layer, which were completely interconnected by feedforward connections. The weights of these connections were initially set to values selected randomly from a uniform distribution between 0 and 1, and normalized such that the sum of all weights feeding into each category unit was 1. Those weights were subsequently modified through training. The model consisted of 100 input units and 63 category units. The 63 input items each contained 100 binary features, and were created at random; each feature had an equal probability of being assigned either a 1 or a 0.

On each training trial the network was presented with a randomly selected (whole) item from the item set, i.e., the item's pattern of features was activated at the input level. The category units computed a weighted sum of this activation based on the connection strength. The weights were modified according to the competitive learning algorithm (Rumelhart & Zipser, 1986):

$$\Delta w_{i,j} = g(c_{i,k}/n_k) - g(w_{i,j}) \quad (1)$$

where $w_{i,j}$ is the weight link connecting the input unit i to the hidden unit j, and g is the learning rate. $c_{i,k}$ is equal to 1 if, in the stimulus pattern k, input unit i is active, and is equal to 0 otherwise. n_k is the total number of active input units in the stimulus pattern k.

Strong lateral inhibition was assumed to operate at the category layer, such that only the most highly activated category unit was allowed to learn. Only the connections that fed to the category unit with the highest activation level (the "winner") were modified: weights to active input units were strengthened, and weights to inactive input units were weakened. The extent of this modification is determined by the value of g, which was set to 0.2 in these simulations. In addition, the activation of each category unit in the simulations reported below was multiplied by a unit-specific sensitivity parameter, also set initially to a random value selected from a uniform distribution between 0 and 1. On each training trial, the sensitivity of the "winning" unit was reduced by a factor of 0.05, and the sensitivities of all "losing" units were increased (equivalently) by a factor necessary to keep the overall amount of sensitivity in the system constant.

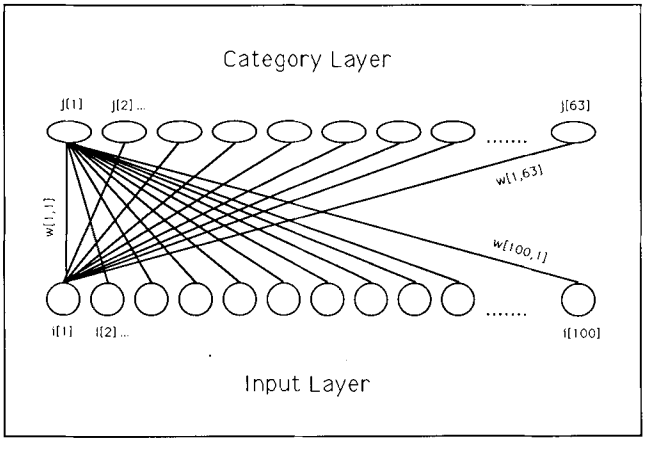

Figure 3. An overview of the competitive-learning model.

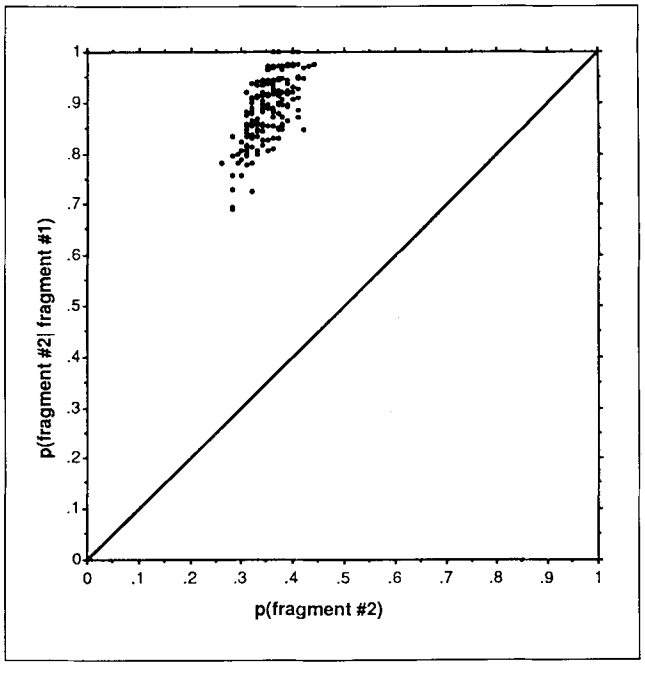

Figure 4. The repeated fragments dependence relation in the competitive network.

To test whether the network had learned to classify the whole items, each item was presented in the same manner as for training, except that no weight modifications were allowed. A given item was deemed to be "correctly identified" on presentation if the winning unit did not win the competition for any other pattern, i.e., if it responded exclusively to that particular item. When the network passed a criterion performance level of 66% correct on the whole items, it was tested for fragment completion.

The first 50 features of each item were present in the first fragment of that item, and features 51 through 100 were assigned a 0. Similarly, the second 50 features of each item were present in the second fragment of that item, while features 1 through 50 were assigned a 0. On presentation of each fragment, the "winning" category unit was noted. A correct completion was tallied if this unit was the same one that coded exclusively for the whole item from which the fragment was constructed. An incorrect completion was recorded otherwise (i.e., if a different category unit responded most strongly to the fragment). The results from the two exclusive fragments of each of the 63 items were tabulated in a two by two contingency table from which the results presented in Figure 4 were computed. The entire procedure was replicated 100 times, giving 100 simulation data points, each based upon 63 observations.

Results

As is shown in Figure 4 the probability of correctly completing the second fragment was dependent on the successful completion of the first. The probability of successful completion of the second fragment, given successful completion of the first, was higher than the probability of completing the second overall.

Discussion

The competitive learning algorithm forces *all* of the weights that connect a winning category unit to active input units to be strengthened, and therefore these weights are linked or bound together. They all rise (or fall) together (though not necessarily by the same quantity). This has two implications. First, it makes the winning category unit more likely to win the competition the next time that same pattern is presented. This is what allows category units to become "dedicated" to a single input pattern, and is indeed at the heart of competitive learning. Second, it produces the dependency seen here with both the independent and the mutually exclusive fragments. This is because a dedicated category unit that responds most strongly to one specific pattern will also respond strongly to both exclusive fragments of that pattern, since all connecting weights used by the pattern as a whole have been strengthened together.

An Autoassociative Back Propagation Model

In this section, we explore a particular nonlinear model—the back propagation autoassociator (Rumelhart, Hinton & Williams, 1986; Cottrell, Munro & Zipser, 1989, see Fig. 5). The autoassociative network is trained to reproduce the input pattern at the output layer. In so doing, it must represent the input in some fashion as a pattern of activation at the hidden layer. The particular representation used depends on the weights on the links. Thus, learning in such networks corresponds to setting these weights. They are learned via an error-correction procedure known as the generalized delta rule (Rumelhart et al., 1986). Activity is propagated as follows: First, the units of the network compute a weighted sum of the activities of the units in the previous layer:

$$\text{net input}_i = \sum_{j=1}^{N} w_{i,j}\,\text{activation}_j \qquad (2)$$

where $\text{activation}_j = \text{input}_j$ if j is an input, and $w_{i,j}$ is the weight from unit j to unit i. Next, a nonlinear "squashing" function is applied to obtain the unit's activity level:

$$\text{activation}_j = 2[1/(1 + e^{-\text{net input}_j})] - 1 \qquad (3)$$

The network is initialized with small random weights. Training proceeds by presenting input vectors, propagating activity through the network using the above equations, producing an activity vector at the output level. This is compared to a desired output vector, the "teaching signal," and an error is computed. The weights are then changed in order to reduce the error at the output level. The particular form of the weight changing rule is

$$\Delta w_{i,j} = -\eta\,\delta_i\,\text{activation}_j \qquad (4)$$

where η is a learning rate, δ_i is the error in the output attributable to unit i, and activation_j is as above. Thus, the weights are set according to the correlation between the error attributable to the unit at one end, and the activity of the unit at the other end. When this equation is applied to the input-to-hidden weights, the activation_j terms are simply the features of the input pattern, and the Δ_i terms are the errors of the hidden units. Thus at the input level, features of the input will be associated with a particular unit at the hidden layer in proportion to the error of that unit on that pattern. To determine whether a particular output is a "hit" or not, a lexicon of patterns was used. The criterion for whether an output pattern matched a lexical item was that the output vector be within half of the distance from that lexical item to the closest other lexical item in the set.

Method

For each trial of the model, a lexicon of 100 vectors,

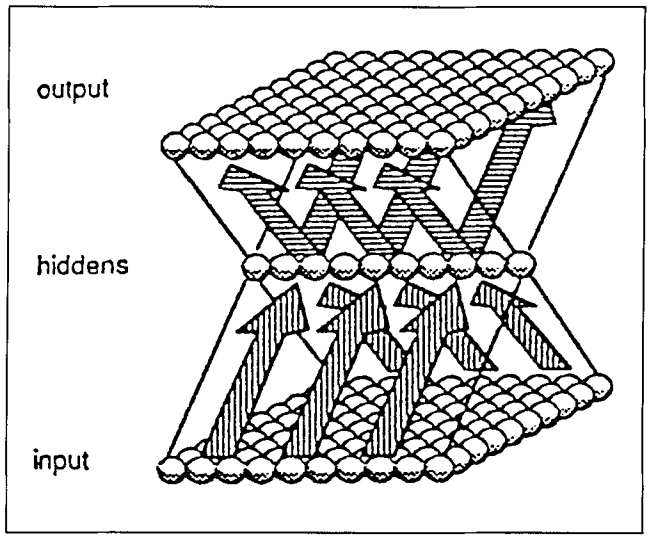

Figure 5. An overview of an autoassociative back propagation model.

each consisting of 64 features, was constructed. The features were determined by a fair coin flip to be either −1 or 1. Thus the items were statistically independent of one another. The model had 50 hidden units, and a learning rate (η) of 0.1 was used. All 100 lexical items were presented to the model, with the desired output patterns (the teaching signal) being identical to the input patterns. The patterns were presented to the model one at a time, the error was computed, and the weights were changed on every presentation. Each simulation run consisted of 10 passes (or epochs) through all 100 patterns.

Two mutually exclusive fragments for each pattern were constructed such that the first 32 features were included in the fragments for test 1; the second 32 were included in the fragments for test 2. These fragments were presented to the trained network without changing the weights. Using the criterion described above, a hit was recorded if the output pattern was within the proper radius of the lexical item from which the fragment was derived. Each trial of 100 new patterns and new random initial weights was repeated 100 times, resulting in 10,000 unique observations.

Results

The results from the two tests were tabulated in a two by two contingency table, from which the simple and conditional probabilities were computed. As shown in Figure 6, the model shows a pattern of positive dependence.

Discussion

With the back propagation model, the individual features of the patterns are linked, internally, via the storage rule. As expressed in Eq. (3), weights from the features (of the same sign) of a pattern will tend to rise or fall

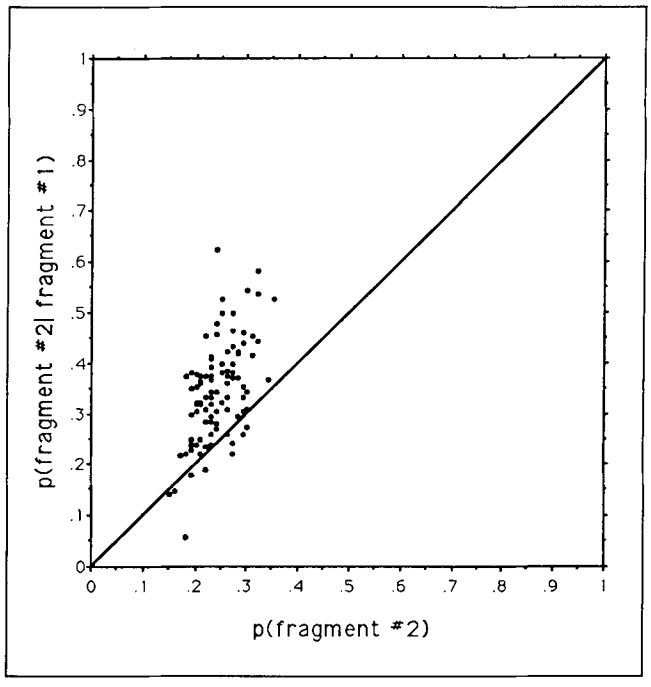

Figure 6. The repeated fragments dependence relation in the back propagation model.

together proportional to the hidden unit's error. That is, the Δ_i term is shared between all of the weight changes to a particular unit. Thus if one feature tends to turn on a hidden unit, so will another feature from the same pattern. Thus the hidden units "bind" the features of a pattern together. Similarly, at the hidden-to-output level, hidden units that are correlated with the error of an output unit will change their weights to that output unit together, so that all of the hidden units responsive to an input pattern will be bound together for that output feature.

A Noninteractive Separate Trace Model: MINERVA

MINERVA (Hintzman, 1986, 1987) is a multiple-trace model of human memory (see Fig. 7). In MINERVA everything—traces, and features within traces—is separate, and as such, the model is noninteractive. This makes it a particularly instructive model, by way of contrast to the three previous models.

Each event is stored as a vector of n features in a separate trace in secondary long-term memory (SM). Each feature is represented as either +1, −1, or 0. During encoding, each feature of the event vector is stored in SM with probability L, the learning rate. If a feature is not stored, a zero is stored as that feature in the SM trace.

Retrieval is simulated by resonating a retrieval cue with all SM traces. This produces a single composite "echo" vector. Since this echo is not the same as the original

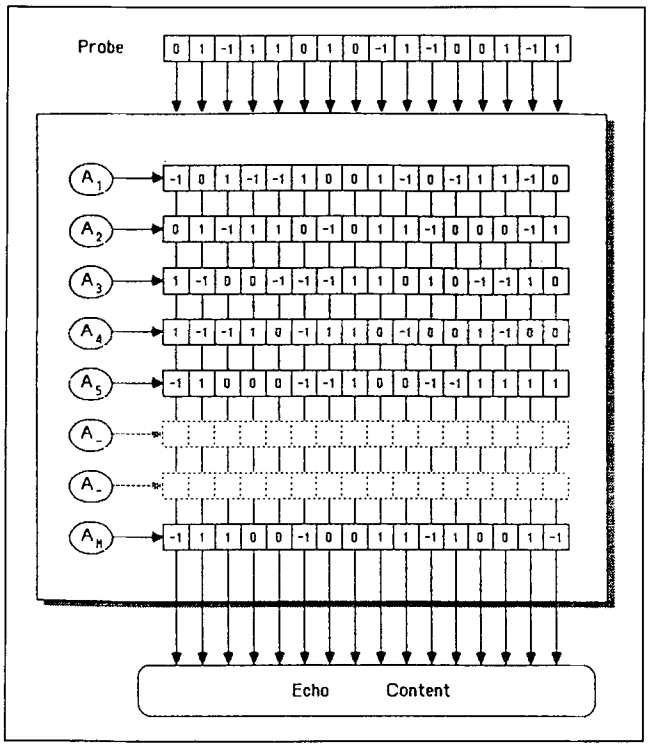

Figure 7. An overview of the MINERVA model.

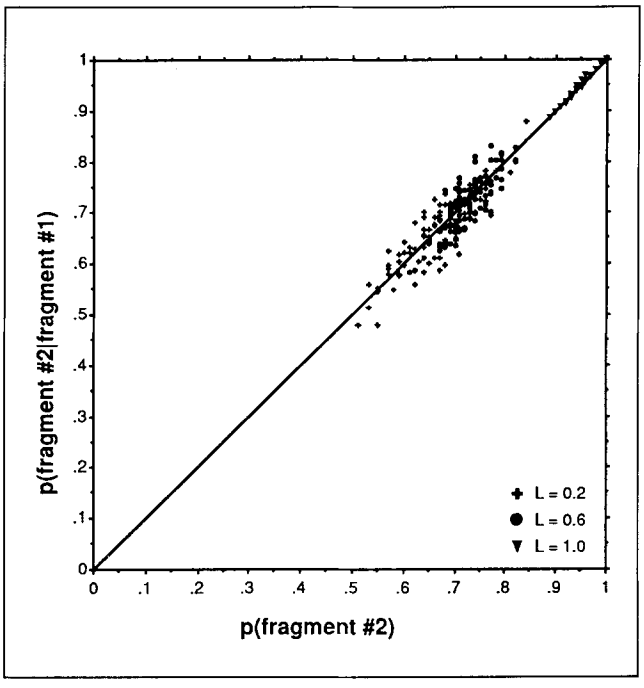

Figure 8. The repeated fragments dependence relation in MINERVA.

patterns, nor even as the SM traces themselves, it is necessary to store all the complete patterns in a lexicon somewhere outside SM. On presentation of a probe, all SM traces (including the missing parts that are not given in the retrieval environment) are assumed to be activated to an extent determined by their similarity to the probe. This similarity, S, is determined by a method similar to that used when computing a Pearson r, except that features that have a value of 0 in either the trace or the probe do not contribute:

$$S_i = \sum_{j=1}^{N} P_j T_{i,j} / N_r \qquad (5)$$

where P_j is the value of feature j in the probe, $T_{i,j}$ is the value of feature j in trace i, and N_r is the number of features that are either a 1 or a -1 in either the trace or the probe. The similarity value is then used to compute the activation of each trace:

$$A_i = S_i^3 \qquad (6)$$

Each trace, weighted by its activation value, is summed into the echo vector. The echo is then matched to every item in a lexicon, and the best match (determined by computing a true Pearson r) is said to be the item produced. A modification of the method Hintzman (1987) used for recall can also be used for the task of fragment completion, as illustrated in the simulations that follow.

Method

A lexicon of 200 items (vectors), each with 20 features, was created with each feature having an equal probability of being either +1, −1, or 0, following the representational coding used by Hintzman (1987). One hundred of these patterns were stored in SM, with a learning rate (L) of 0.2, 0.6, or 1.0.

The model was tested on two mutually exclusive fragments of each pattern, with an equal number of features in each fragment. The first 10 features of each original pattern were assigned to the first fragment of that pattern, while features 11 through 20 were assigned to the second fragment. Features of the original pattern that were not present in the fragment were designated as 0. Each fragment was presented as a probe that was compared to the 100 stored vectors in SM, and the complete resulting echo was correlated with each of the 200 patterns in the lexicon. If the pattern giving the highest r was the same one used to generate the fragment, then a correct completion was tallied. This procedure was replicated 50 times, and the simple and conditional probabilities of successful completion were computed bidirectionally [i.e., both p(fragment #2|fragment #1) and p(fragment #1|fragment #2) for each pair of fragments]. Thus, each of the 100 (50 × 2) data points is based on 100 simulated observations.

Results

As is shown in Figure 8, MINERVA produces indepen-

dence. This is just the right pattern to account for the implicit memory data.

CONCLUSION

The three distributed/interactive models produce the dependence results, as shown in the explicit memory data, whereas the noninteractive separate-trace model MINERVA produces the independence results, as shown in the implicit memory data. The critical difference between the two kinds of models is that those producing dependence have some inherent mechanism that fuses or glues the elements together in the items that are processed. The separate trace model does not have this characteristic. In that model the features are simply lined up in a vector but there is no operation or modification that alters any of them or the connections to them (at storage) as a function of the values of the others. They are inert with respect to one another—noninteracting—and, as a result, the features within the separate-trace model maintain their functional (and statistical!) independence. Those models that glue, fuse, or bind together the diverse elements of the events presented to them to form a coherent whole produce the dependence findings shown by human subjects under explicit memory instructions. Those models that do not so bind the features together show independence, as is shown in the human implicit-memory data.

Much discussion has been devoted to the idea that a critical distinction between explicit and the implicit memory tasks is the role played by consciousness, or attention. The episodic system is assumed to require consciousness, and to be distinguished from nonepisodic (or Quasi Memory) system on that basis. But we may still ask, what does conscious attention do, or what is the functional difference between systems that do or do not require it? If we subscribe to the idea that involvement of consciousness is a distinguishing characteristic of the two systems (and that the models exhibit a critical functional distinction) then we may say that conscious attention is a prerequisite for that which distinguishes the models: the binding function.

Within a somewhat different research framework, Treisman and Gelade (1980) also see consciousness or attention as a necessary gateway into the "true" memory system, although certainly some processing can be done without it. According to Treisman's view, prior to focused attention the various features in the perceptual world are free floating rather than integrated—the supporting evidence being the discovery of *illusory conjunctions*. For example, given a blue circle and a red square presented very quickly, a person might report seeing a red circle. The role of conscious attention, then, is to fuse the elements in an event together correctly.

We did not set out to find a connection between the dependence and independence data on fragment completion as described by Hayman and Tulving and the work of Treisman and her colleagues on illusory conjunctions. Nevertheless, the manner in which the models explain the implicit-memory and the explicit-memory data makes sense in terms of these other ideas, and offers support for them. As we have shown, what the distributed models do (accounting for the explicit memory data), that the separate-trace model does not do (and hence accounts for the implicit memory data), is fuse the episodically presented features of events together to form a cohesive whole—a whole in which the parts will and do, if tested, exhibit interdependence.

Acknowledgments

We gratefully acknowledge support of the National Institute of Mental Health Grant 1R29MH48066-01 to JM. Thanks go to Endel Tulving, David Zipser, and Bennett Schwartz.

REFERENCES

Anderson, J. R. (1978). Arguments concerning representations for mental imagery. *Psychological Review, 85*, 249–277.

Cottrell, G. W., Munro, P., & Zipser, D. (1989). Image compression by back propagation: An example of extensional programming. In N. Sharkey (Ed.), *Models of cognition: A review of cognitive science* (Vol. 1). Norwood: Ablex.

Flexser, A. J. (1981). Homogenizing the 2 × 2 contingency table: A method for removing dependencies due to subject and item differences. *Psychological Review, 88*, 327–339.

Graf, P., Mandler, G., & Haden, P. (1982). Simulating amnesic symptoms in normal subjects. *Science, 218*, 1243–1244.

Grossberg, S. (1970). Some networks that can learn, remember, and reproduce any number of complicated space-time patterns. *Studies in Applied Mathematics, 49*, 135–166.

Hayman, C. A. G., & Tulving, E. (1989a). Is priming in fragment completion based on a "Traceless" memory system? *Journal of Experimental Psychology: Learning, Memory and Cognition, 15*, 941–956.

Hayman, C. A. G., & Tulving, E. (1989b). Contingent dissociation between recognition and fragment completion: The method of triangulation. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*, 228–240.

Hintzman, D. L. (1986). "Schema abstraction" in a multiple-trace memory model. *Psychological Review, 93*, 411–428.

Hintzman, D. L. (1987). Recognition and recall in MINERVA2: Analysis of the "recognition-failure" paradigm. In P. Morris (Ed.), *Modeling cognition* (pp. 215–229). New York: John Wiley.

Jacoby, L. L., & Dallas, M. (1981). On the relationship between autobiographical memory and perceptual learning. *Journal of Experimental Psychology: General, 110*, 306–340.

Metcalfe, J. (1991). Recognition failure and the composite memory trace in CHARM. *Psychological Review, 98*, 529–553.

Metcalfe, J. (1992). Monitoring and control in a composite holographic associative recall model (CHARM): Implications for Korsakoff amnesia. *Psychological Review*, in press.

Metcalfe Eich, J. (1985). Levels of processing, encoding specificity, elaboration, and CHARM. *Psychological Review, 92*, 1–38.

Richardson-Klavehn, A., & Bjork, R. A. (1988). Measures of memory. *Annual Review of Psychology, 39*, 475–543.

Roediger, H. L., III, Weldon, M. S., & Challis, B. H. (1989). Explaining dissociations between implicit and explicit measures of retention: A processing account. In H. L. Roediger III & F. I. M. Craik (Eds.), *Varieties of memory and consciousness, Essays in honour of Endel Tulving* (pp. 3–41). Hillsdale, NJ: Erlbaum.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature (London), 323*, 533–536.

Rumelhart, D. E., & Zipser, D. (1986). Feature discovery by competitive learning. In D. E. Rumelhart & J. L. McClelland (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition* Vol. 1 (pp. 151–193). Cambridge: MIT Press.

Schacter, D. L. (1987). Implicit memory: History and current status. *Journal of Experimental Psychology: Learning, Memory and Cognition, 13*, 501–518.

Shimamura, A. P. (1985). Problems with the finding of stochastic independence as evidence for the independence of cognitive processes. *Bulletin of the Psychonomic Society, 23*, 506–508.

Shimamura, A. P. (1986). Priming effects in amnesia: Evidence for a dissociable memory function. *Quarterly Journal of Experimental Psychology, 38A*, 619–644.

Treisman, A., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology, 12*, 97–136.

Tulving, E. (1986). What kind of hypothesis is the distinction between episodic and semantic memory? *Journal of Experimental Psychology: Learning, Memory and Cognition, 12*, 307–311.

Tulving, E., Schacter, D. L., & Stark, H. A. (1982). Priming effects in word fragment completion are independent of recognition memory. *Journal of Experimental Psychology: Learning, Memory and Cognition, 8*, 336–342.

von der Malsberg, C. (1973). Self-organizing of orientation sensitive cells in the striate cortex. *Kybernetic, 14*, 85–100.

Watkins, M. J., & Todres, A. K. (1978). On the relation between recall and recognition. *Journal of Verbal Learning and Verbal Behavior, 17*, 621–633.

Witherspoon, D., & Moscovitch, M. (1989). Stochastic independence between two implicit memory tasks. *Journal of Experimental Psychology: Learning, Memory and Cognition, 15*, 22–30.